

Ponavljjanje nastavnih sadržaja iz informatike

TJEDAN PONAVLJANJA I PLANIRANJA

Što trebate ponavljati?

Dragi učenici šestih razreda, ovaj tjedan ćemo ponavljati nastavno gradivo iz informatike koje se odnosi na nastavnu temu **5.B. Programski jezik Python**. Ova nastavna tema započinje na stranici 68 u udžbeniku, a završava na stranici 87 u udžbeniku.

Preporučam da pročitate i proučite tekst koji se odnosi na tu nastavnu temu, ali s obzirom na izvanredne okolnosti i teškoću gradiva ja ću vam na sljedećim stranicama ovog dokumenta napisati i objasniti što trebate znati. Ovu nastavnu temu ću ocjenjivati!

Na zadnjoj stranici ovog dokumenta nalazi se pet zadataka koje trebate riješiti i poslati mi danas u obliku privatne poruke na Yammeru.

Što trebate znati?

1. Trebate znati osnovne računске operacije u Pythonu, a osobito pripazite na računsku operaciju dijeljenja (imamo dijeljenje koje uvijek vraća cijeli broj i dijeljenje koje uvijek vraća decimalni broj):

```
Python Shell
File Edit Shell Debug Options Windows Help
>>> 2 + 3
5
>>> 13 - 6
7
>>> 7 * 5
35
>>> 45 / 2
22.5
>>> 45 // 5
9
>>> |
```

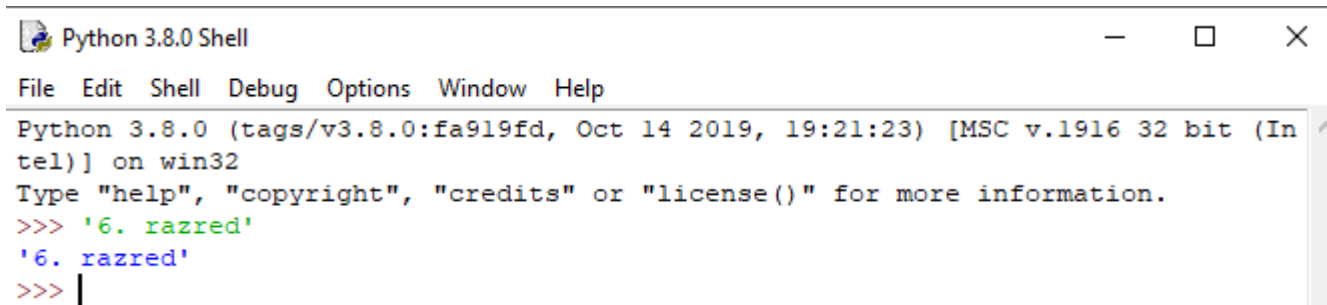
računska operacija	znak	primjer	
		operacija	rezultat
Zbrajanje	+	2 + 3	5
Oduzimanje	-	13 - 6	7
Množenje	*	7 * 5	35
Dijeljenje	/	45 / 2	22.5
		45 / 5	9.0
djelomični količnik	//	45 // 5	9
		14 // 3	4

Vraća decimalni broj

Vraća cijeli broj

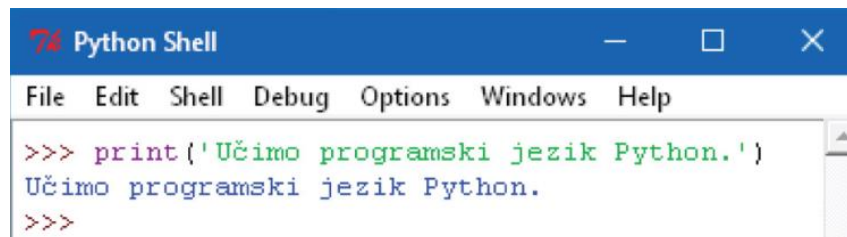
Što trebate znati?

2. Trebate znati da se znakovni nizovi (tekst) uvijek pišu **unutar navodnika** (može i unutar jednostrukih i unutar dvostrukih, ali se preporuča unutar jednostrukih):



```
Python 3.8.0 Shell
File Edit Shell Debug Options Window Help
Python 3.8.0 (tags/v3.8.0:fa919fd, Oct 14 2019, 19:21:23) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> '6. razred'
'6. razred'
>>> |
```

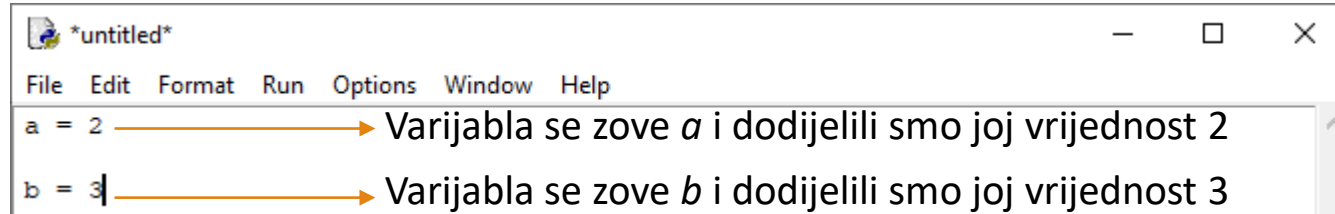
3. Trebate znati da se naredba **print** koristi za ispis neke vrijednosti na zaslon! Ta vrijednost može biti broj, tekst, vrijednost varijable... Primjer ispisa teksta na zaslon pomoću naredbe **print**:



```
Python Shell
File Edit Shell Debug Options Windows Help
>>> print('Učimo programski jezik Python.')
Učimo programski jezik Python.
>>>
```

Što trebate znati

4. Trebate znati da se varijable definiraju tako da napišemo ime varijable i dodijelimo joj neku vrijednost. Znak = služi za dodjeljivanje neke vrijednosti varijabli. Primjer:



```
*untitled*
File Edit Format Run Options Window Help
a = 2
b = 3
```

5. Trebate znati napisati jednostavan program za obavljanje neke računске operacije s varijablama. Evo primjera jednog takvog programa koji zbraja vrijednosti dviju varijabli i rezultat ispisuje na zaslon pomoću naredbe **print**:

Ova naredba print će ispisati tekst $a = i$ desno od toga vrijednost varijable a ; dakle ispisat će $a = 80$. Tekst i varijabla su odvojeni **zarezom** u toj print naredbi.

```
a = 80
b = 30
zbroj = a + b
print('a = ', a)
print('b = ', b)
print('zbroj = ', zbroj)
```

Što trebate znati?

6. Trebate znati da se naredba `input` koristi za upis podataka s tipkovnice. Kada pokrenemo program i kada Python naiđe na naredbu `input`, on će stati i čekati nas da nešto upišemo na tipkovnici. Kada pritisnemo Enter, Python će to što smo upisali spremiti u neku varijablu i nastaviti s izvršavanjem programa. Unutar zagrada naredbe `input` možemo napisati nekakvu poruku koja će se prikazati prije nego što krenemo unositi podatke s tipkovnice.

Primjer programa koji pomoću naredbe `input` omogućava unos podatka s tipkovnice, sprema ga u varijablu, pretvara ga u cijeli broj i ispisuje ga na zaslon.

```
a = input('Upišite neki broj:')  
a = int(a)  
print(a)
```

Naredba `input` uneseni podatak tumači kao tekst. Ako želimo taj podatak koristiti kao broj onda ga moramo pretvoriti u broj pomoću naredbe `int`.

Što trebate znati?

7. Trebate znati napisati jednostavan program koji koristi naredbe odluke. To su naredbe **if**, **elif**, **else**. Promotrite sljedeći primjer:

```
a = 5
if a > 5:
    print('a je veće od 5')
elif a < 5:
    print('a je manje od 5')
else:
    print('a je jednako 5')
```

Ova print naredba će se izvršiti samo ako je vrijednost varijable a veća od 5. U tom slučaju ostale naredbe se preskaču!

Ova print naredba će se izvršiti ako vrijednost varijable a nije veća od 5 i ako je vrijednost varijable a manja od 5. U tom slučaju preostale naredbe se preskaču!

U ovom programu izvršit će se samo jedna print naredba, i to ova **zadnja** jer uvjeti $a > 5$ i $a < 5$ nisu istiniti!

Ova print naredba će se izvršiti ako nijedan od gornjih uvjeta nije istinit, odnosno izvršit će se ako vrijednost varijable a nije ni veća ni manja od 5, nego je jednaka 5.

Što trebate znati?

8. Naučite i sljedeće znakove za uspoređivanje kod naredbi odluka (**if**, **elif**, **else**). Posebnu pažnju obratite na dva znaka jednako (`==`). Dakle ako želimo provjeriti jesu li dvije vrijednosti jednake, koristimo dva znaka jednako (`==`).

Matematika	Python
<	<
>	>
=	==
≥	>=
≤	<=
≠	!=

Što trebate znati?

9. Trebate znati što će ispisivati sljedeće petlje:

```
for k in range(5):  
    print(k)
```

Ispisat će sljedeće brojeve: 0, 1, 2, 3 i 4.
Ako je samo jedan broj u funkciji range, onda krećemo od 0, idemo do broja koji je za 1 manji u odnosu na broj u funkciji range ($5 - 1 = 4$), i pri svakom prolazu petlje broj se povećava za 1.

```
for k in range(2, 6):  
    print(k)
```

Ispisat će sljedeće brojeve: 2, 3, 4 i 5.
Ako su dva broja u funkciji range, onda krećemo od prvog broja iz funkcije range (u ovom slučaju to je 2), idemo do broja koji je za 1 manji u odnosu na zadnji broj u funkciji range ($6 - 1 = 5$), i pri svakom prolazu petlje broj se povećava za 1.

Što trebate znati

Nastavak od prethodne stranice:

```
for k in range(4, 17, 4):  
    print(k)
```

Ispisat će sljedeće brojeve: 4, 8, 12 i 16.
Ako su tri broja u funkciji range, onda krećemo od prvog broja iz funkcije range (u ovom slučaju to je 4), idemo do broja koji je za 1 manji u odnosu na srednji broj u funkciji range ($17 - 1 = 16$), i pri svakom prolazu petlje broj se povećava za 4 (zadnji broj u funkciji range – to je korak petlje).

Što trebate znati?

10. Petlja s logičkim uvjetom, odnosno petlja u kojoj se koristi riječ *while*. Ova petlja će izvršavati naredbe unutar sebe sve dok je istinit uvjet koji se nalazi desno od riječi *while*. Primjer:

```
a = 0
while a < 5:
    print(a)
    a = a + 1
```

Naredbe unutar petlje (a to su $print(a)$ i $a = a + 1$) će se izvršavati sve dok je uvjet $a < 5$ istinit, odnosno dok je vrijednost varijable a manja od 5. Budući da se u svakom koraku petlje varijabla a povećava za 1 (to je naredba $a = a + 1$), u jednom trenutku vrijednost varijable a neće biti manja od 5 pa će petlja prekinuti s izvršavanjem.

Zadaci

1. Koji će biti rezultat (što će se ispisati) izvršavanja sljedeće računske operacije u Pythonu: $9 / 3$
2. Napišite naredbu koja će na zaslon ispisati tekst *Python je programski jezik*.
3. Napišite naredbu kojom ćete definirati varijablu *d* i pridružite joj vrijednost 7.
4. Pretpostavimo da imamo varijablu *a* kojoj je dodijeljena vrijednost 2. Napišite naredbe odluke (if, elif, else) koje će ispisati tekst *a je veće od 2* ako je varijabla *a* veća od 2, zatim tekst *a je manje od 2* ako je varijabla *a* manja od 2 i tekst *a je jednako 2* ako nijedan od spomenuta dva uvjeta nije istinit.
5. Koje će brojeve ispisati sljedeća petlja?

```
for k in range(10, 20, 2):  
    print(k, end= ' ')
```